CHAPTER 3
Frequency Distribution, Central Tendency, Variability

Introduction

Descriptive statistics is a way of better describing and summarizing the data and its characteristics, in a meaningful way. The part of descriptive statistics includes the measure of frequency distribution, the measure of central tendency, which includes mean, median, mode, measure of variability, measure of association, and shapes. Descriptive statistics simply show what the data shows. Frequency distribution is primarily used to show the distribution of categorical or numerical observations, counting in different categories and ranges. Central tendency calculates the mode, which is the most frequent data set, median which is the middle value in an ordered set and mean which is the average value. The measures of variability estimate how much the values of a variable are spread, or it calculates the variations in the value of the variable. They allow us to understand how far the data deviate from the typical or average value. Range, variance,

and standard deviation are commonly used measures of variability. Measures of association estimate the relationship between two or more variables, through scatterplots, correlation, regression. Shapes describe the pattern and distribution of data by measuring skewness, symmetry of shape, bimodal, unimodal, and uniform modality, kurtosis, counting and grouping.

Structure

In this chapter, we will discuss the following topics:

- Measures of frequency
- Measures of central tendency
- Measures of variability or dispersion
- Measures of association
- Measures of shape

Objectives

By the end of this chapter, readers will learn about descriptive statistics and how to use them to gain meaningful insights. You will gain the skills necessary to calculate measures of frequency distribution, central tendency, variability, association, shape, and how to apply them using Python.

Measure of frequency

A measure of frequency counts the number of times a specific value or category appears within a dataset. For example, to find out how many children in a class like each animal, you can apply the measure of frequency on a data set that contains the five most popular animals. *Table 3.1* displays how many times each animal was chosen by the 10 children. Out of the 10 children, 4 like dogs, 3 like cats, 2 like cow, and 1 like rabbit.

Animal	Frequency
Dog	4
Cat	3
Cow	2
Rabbit	1

Table 3.1: *Frequency of animal chosen by children*

Another option is to visualize the frequency using plots, graphs, and charts. For example, we can use pie chart, bar chart, and other charts.

Tutorial 3.1: To visualize the measure of frequency using pie chart, bar chart, by showing both plots in subplots, is as follows:

```
import pandas as pd
import matplotlib.pyplot as plt
# Create a data frame with the new data
data = {"Animal": ["Dog", "Cat", "Cow", "Rabbit"],
        "Frequency": [4, 3, 2, 1]}
df = pd.DataFrame(data)
# Create a figure with three subplots
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(18, 6))
# Plot a pie chart of the frequency of each animal on the
 first subplot
ax1.pie(df["Frequency"], labels=df["Animal"], autopct="
%1.1f%%")
ax1.set_title("Pie chart of favorite animals")
# Plot a bar chart of the frequency of each animal on the
second subplot
ax2.bar(df["Animal"], df["Frequency"], color=
["brown", "orange", "black", "gray"])
ax2.set_title("Bar chart of favorite animals")
```

```
15. ax2.set_xlabel("Animal")
16. ax2.set_ylabel("Frequency")
17. # Save and show the figure
18. plt.savefig('measure_frequency.jpg',dpi=600,bbox_inches='tight')
19. plt.show()
```

Output:

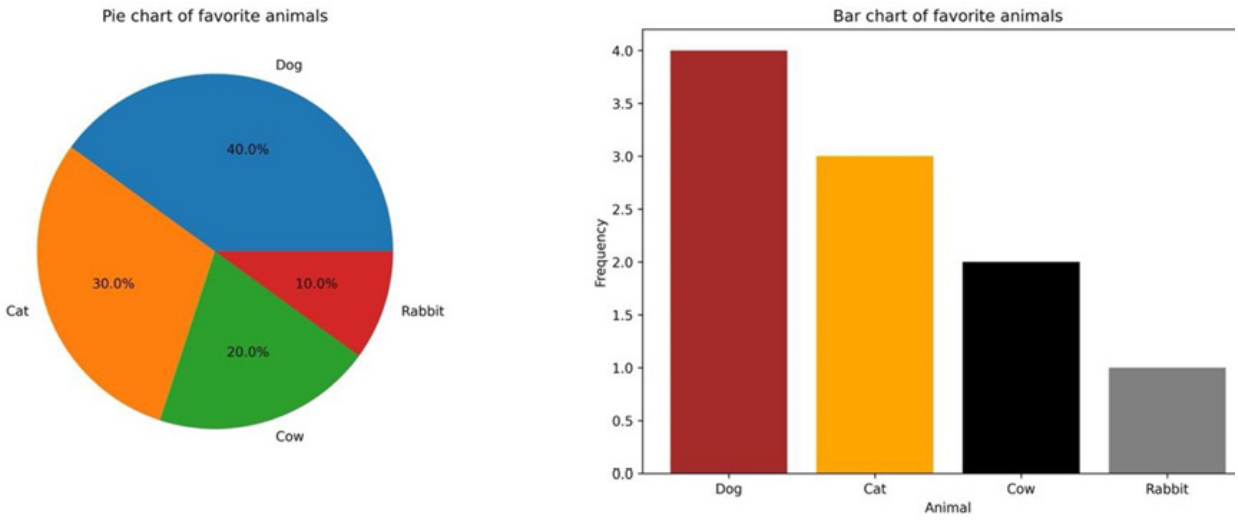

Figure 3.1: *Frequency distribution in pie and bar charts*

Frequency tables and distribution

Frequency tables and distribution are methods of sorting and summarizing data in descriptive statistics. **Frequency tables** display how often each value or category of a variable appears in a dataset. **Frequency distribution** exhibits the frequency pattern of a variable, which can be illustrated using graphs or tables. Distribution is a way of summarizing and displaying the number or proportion of observations for each possible value or category of a variable.

For example, on the data about favorite animals of ten school children, you can create a table that displays how many children like each animal and a distribution chart that

reveals the data's shape as discussed above in the measure of frequency and in the examples of relative and cumulative frequency, as explained in the next section.

Relative and cumulative frequency

Relative frequency is the ratio of the number of times a value or category appears in the data set to the total number of data values. Its relative and calculated by dividing the frequency of a category by the total number of observations. On the other hand, **cumulative frequency** is the total number of observations that fit into a specific range of categories, along with all of the categories that came before it. To calculate it, add the frequency of the current category to the cumulative frequency of the previous category.

For example, suppose we have a data set of the favorite animals of 10 children, as shown in the *Table 3.1* above. To determine the relative frequency of each animal, divide the frequency by the total number of children, which is 10. Doing so for dogs the relative frequency is 4/10 = 0.4, meaning that 40% of the children like dogs. For cats, it is *3/10 = 0.3*, meaning that 30% of the children like cats. Further relative frequencies of each animal are shown in the following table:

Animal	Frequency	Relative frequency
Dog	4	0.4
Cat	3	0.3
Cow	2	0.2
Rabbit	1	0.1

Table 3.2: *Relative frequency of each animal*

Now, to calculate the cumulative frequency for each animal, add up the relative frequencies of all animals that are less

than or equal to the current animal in the table. For example, dog's cumulative frequency is 0.4, identical to their relative frequency. The cumulative frequency of cats is *0.4 + 0.3 = 0.7*, indicating that 70% of the children prefer dogs or cats. Similarly, relative frequency of cow is *0.4 + 0.3 + 0.2 = 0.9*, which means 90% of the children like dogs, cats and cow, as in shown in *Table 3.3*:

Animal	Frequency	Relative frequency	Cumulative relative frequency
Dog	4	0.4	0.4
Cat	3	0.3	0.7
Cow	2	0.2	0.9
Rabbit	1	0.1	1

Table 3.3: *Comparison of relative and cumulative relative frequency*

Tutorial 3.2: An example to view the relative frequency in pie chart and cumulative frequency in a line plot, is as follows:

```
1. import pandas as pd
2. import matplotlib.pyplot as plt
3. # Create a data frame with the given data
4. data = {"Animal": ["Dog", "Cat", "Cow", "Rabbit"],
5.         "Frequency": [4, 3, 2, 1]}
6. df = pd.DataFrame(data)
7. # Calculate the relative frequency by dividing the freque
   ncy by the sum of all frequencies
8. df["Relative Frequency"] = df["Frequency"] / df["Freque
   ncy"].sum()
9. # Calculate the cumulative frequency by adding the rela
   tive frequencies of all the values that are less than or eq
   ual to the current value
```

```
df["Cumulative Frequency"] = df["Relative Frequency"].cumsum()
# Print the data frame with the relative and cumulative frequency columns
print(df)
# Create a figure with two subplots
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(12, 6))
# Plot a pie chart of the relative frequency of each animal on the first subplot
ax1.pie(df["Relative Frequency"], labels=df["Animal"], autopct="%1.1f%%")
ax1.set_title("Pie chart of relative frequency of favorite animals")
# Plot a line chart of the cumulative frequency of each animal on the second subplot
ax2.plot(df["Animal"], df["Cumulative Frequency"], marker="o", color="red")
ax2.set_title("Line chart of cumulative frequency of favorite animals")
ax2.set_xlabel("Animal")
ax2.set_ylabel("Cumulative Frequency")
# Show the figure
plt.savefig('relative_cummalative.jpg',dpi=600,bbox_inches='tight')
plt.show()
```

Output:

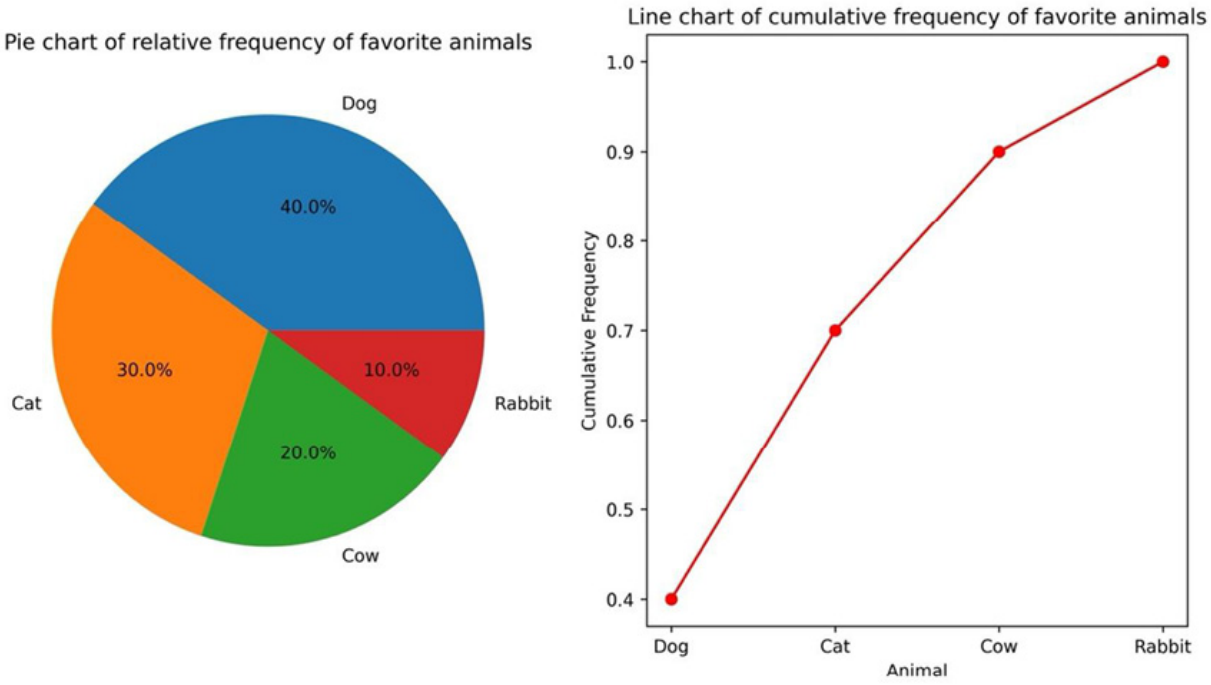

Figure 3.2: *Relative frequency in pie chart and cumulative frequency in a line plot*

Measure of central tendency

Measure of central tendency is a method to summarize a data set using a single value that represents its center or typical value. This helps us understand the basic features of the data and compare different sets. There are three common measures of central tendency: the mean, the median, and the mode. The average, or mean, is found by adding up all the numbers and then dividing by the total length of numbers. For example, let us say we have five test scores: 80, 85, 90, 95, and 100. To find the mean, we add up all the scores and divide by 5. This gives us the following:

$$(80 + 85 + 90 + 95 + 100) / 5 = 90.$$

The median is the middle number when all the numbers are arranged in order, either from smallest to largest or largest to smallest. To calculate the median, we start by organizing the data and selecting the value in the middle. If the data

set has an even number of values, we average the two middle values. For instance, if there are five test scores, 80, 85, 90, 95, and 100, the median is 90, since it is the third value in the sorted list. If we have six test scores, 80, 85, 90, 90, 95, and 100, the median is the average of 90 and 90, which is also 90. The center number in a set of numbers is called the median. To find the number that appears most often in a set, we count how many times each number appears. In a set of five scores: 80, 85, 80, 95, and 100, the mode is 80 since it appears more than once. However, in a set of six scores: 80, 85, 90, 90, 95, and 100, the mode is 90 since it appears twice, which is more frequent. If all numbers appear the same number of times, there is no mode. We also discussed mean, median, and mode measures in *Chapter 2, Exploratory Data Analysis.*

Let us recall the measure of central tendency with an example to compute the salary in different regions of Norway, based on the average income by region.

The following table shows the data:

Region	Oslo	South	Mid-Norway	North
Salary (NOK)	57,000	54,000	53,000	50,000

Table 3.4: Average income by region in Norway

To find the middle value, average, and the most frequent value in this set of salaries, we can use the median, mean, and mode, respectively. The mean is the sum of all the salaries divided by 4, which equals *(57,000 + 54,000 + 53,000 + 50,000) / 4 = 53,500*. The two middle numbers are 54,000 and 53,000. We can calculate the median by adding up the four numbers and dividing the sum by 2. As a result, the middle value of the 4 numbers is 53,500. In this case, none of the salaries have the same frequency, hence there is no mode.

Tutorial 3.3: Let us look at an example to compute the

measure of central tendency with a python function. Refer to the following table:

Country	Salary (NOK)
USA	57,000
Norway	54,000
Nepal	50,000
India	50,000
China	50,000
Canada	53,000
Sweden	53,000

Table 3.5: *Salary in different countries*

Code:

```
import pandas as pd
import statistics as st
# Define a function that takes a data frame as an argument and returns the mean, median, and mode of the salary column
def central_tendency(df):
    # Calculate the mean, median, and mode of the salary column
    mean = df["Salary (NOK)"].mean()
    median = df["Salary (NOK)"].median()
    mod = st.mode(df["Salary (NOK)"])
    # Return the mean, median, and mode as a tuple
    return (mean, median, mod)
# Create a data frame with the new data
data = {"Country": ["USA", "Norway", "Nepal", "India", "China", "Canada", "Sweden"],
        "Salary (NOK)": [57000, 54000, 50000, 50000, 500
```

```
00, 53000, 53000]}
df = pd.DataFrame(data)
# Call the function and print the results
mean, median, mod = central_tendency(df)
print(f"The mean of the salary is {mean} NOK.")
print(f"The median of the salary is {median} NOK.")
print(f"The mode of the salary is {mod} NOK.")
```

Output:

```
The mean of the salary is 52428.57142857143 NOK.
The median of the salary is 53000.0 NOK.
The mode of the salary is 50000 NOK.
```

Measures of variability or dispersion

Measures of variability is a measure that show how spread-out data is from the center or scattered a set of data points are. They help to summarize and understand the data better. Simply, measures of variability help you figure out if your data points are tightly packed around the average or spread out over a wider range. Measuring variability or dispersion is important for several reasons as follows:

- It is simpler to compare various data sets thanks to their ability to quantify variability. We can determine that one group of data is more variable or distributed than the other if, for example, the two sets have the same average but different ranges.
- They help determine the form and features of the distribution. For example, a high degree of variation in the data could indicate skewness or outliers. A low degree of variability in the data may indicate that it is normal or symmetric.
- They help in testing hypothesis and using data to guide decisions. For example, when there is little variability in the data, the sample better represents the whole group,

resulting in more comprehensive and reliable conclusions. On the other hand, when there is a high degree of variability, the sample is not as representative of the population, leading to less trustworthy conclusions.

Some common measures of variability or dispersion are, range, variance, standard deviation, interquartile range. Range is the difference between the highest and lowest values in a data. For example, if you have a dataset with numbers: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, the range would be 9 (the difference of the highest and the lowest score).

Tutorial 3.3: An example to compute the range in the data, is as follows:

```
# Define a data set as a list of numbers
data = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
# Find the maximum and minimum values in the data set
max_value = max(data)
min_value = min(data)
# Calculate the range by subtracting the minimum from the maximum
range = max_value - min_value
# Print the range
print("Range:", range)
```

Output:

```
Range: 9
```

Interquartile range (IQR) is difference between third and first quartile of a data, which measures spread of the middle 50% of the data. For example, lets compute the IQR of the data set 1, 2, 3, 4, 5, 6, 7, 8, 9, 10.

First quartile *(Q1) = 3.25*

Third quartile *(Q3) = 7.75*

Then *IQR = Q3 – Q1 = 7.75 – 3.25 = 4.5*

Tutorial 3.4: An example to compute the interquartile range in data, is as follows:

```
import numpy as np
# Dataset
data = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
# Calculate the first quartile (Q1)
q1 = np.percentile(data, 25)
# Calculate the third quartile (Q3)
q3 = np.percentile(data, 75)
# Calculate the interquartile range (IQR)
iqr = q3 - q1
print(f"Interquartile range:: {iqr}")
```

Output:

```
Interquartile range: 4.5
```

Variance equals the mean of the squared distances between data points. For example, in a set of 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, where the mean is 5.5, the variance would be 8.25.

Tutorial 3.5: An example to compute the interquartile range in data, is as follows:

```
import statistics
# Define a data set as a list of numbers
data = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
# Find the mean of the data set
mean = statistics.mean(data)
# Find the sum of squared deviations from the mean
ssd = 0
for x in data:
    ssd += (x - mean) ** 2
# Calculate the variance by dividing the sum of squared deviations by the number of values
```

```
variance = ssd / len(data)
print("Variance:", variance)
```

Output:

```
Variance: 8.25
```

Standard deviation is square root of variance which measures how much data points deviate from mean. For example, in a data of 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 standard deviation is 2.87.

Tutorial 3.6: An example to compute the standard deviation in data, is as follows:

```
# Import math library
import math
# Define a data set
data = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
# Find the mean of the data set
mean = sum(data) / len(data)
# Find the sum of squared deviations from the mean
ssd = 0
for x in data:
    ssd += (x - mean) ** 2
# Calculate the variance by dividing the sum of squared deviations by the number of values
variance = ssd / len(data)
# Calculate the standard deviation by taking the square root of the variance
std = math.sqrt(variance)
print("Standard deviation:", std)
```

Output:

```
Standard deviation: 2.87
```

Mean deviation is the average of the absolute distances of each value from the mean, median or mode. For example, in

a data of 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 the mean deviation is 2.5.

Tutorial 3.7: An example to compute the mean deviation in data, is as follows:

```
# Define a data set as a list of numbers
data = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
# Calculate the mean of the data set
mean = sum(data) / len(data)
# Calculate the mean deviation by summing the absolute differences between each data point and the mean
mean_deviation = sum(abs(x - mean) for x in data) / len(data)
# Print the mean deviation
print("Mean Deviation:", mean_deviation)
```

Output:

```
Mean Deviation: 2.5
```

Measure of association

Measure of association is used to describe how multiple variables are related to each other. The measure of association varies and depends on the nature and level of measurement of variables. We can measure the relationship between variables by evaluating their strength and direction of association while also determining their independence or dependence through hypothesis testing. Before we go any further, let us understand what hypothesis testing is **Hypothesis testing** is used in statistics to investigate ideas about the world. It's often used by scientists to test certain predictions (called hypotheses) that arise from theories. There are two types of hypotheses: null hypotheses and alternative hypotheses. Let us understand them with an example where a researcher want to see, if there is a relationship between gender and height. Then the hypotheses are as follows.

- **Null hypothesis (H_0)**: States the prediction that there is no relationship between the variables of interest. So, for the example above, the null hypothesis will be that men are not, on average, taller than women.
- **Alternative hypothesis (H_a or H_1)**: Predicts a particular relationship between the variables. So, for the example above, the alternative hypothesis to null hypothesis will be that men are, on average, taller than women.

Continuing measures of association, it can help identify potential causal factors, confounding variables, or moderation effects that impact the outcome in question. Covariance, correlation, chi-squared, Cramer's V, and contingency coefficients, discussed below, are used in statistical analyses to understand the relationships between variables.

To demonstrate the importance of a measure of association, let us take a simple example. Suppose we wish to investigate the correlation between smoking habits and lung cancer. We collect data from a sample of individuals, recording whether or not they smoke and whether or not they have lung cancer. Then, we can employ a measure of association, like the chi-square test (described further below), to ascertain if there is a link between smoking and lung cancer. The chi-square test assesses the extent to which smoking, and lung cancer frequencies observed differ from expected frequencies, assuming their independence. A high chi-square value demonstrates a notable correlation between the variables, while a low chi-square value suggests that they are independent.

For example, suppose we have the following data, and we want to see the effect of smoking in lung cancer:

Smoking	Lung Cancer	No Lung Cancer	Total
Yes	80	20	100

No	20	80	100
Total	100	100	200

Table 3.6: *Frequency of patients with and without cancer of the lung and their smoking habits*

Based on *Table 3.6*, we can calculate the observed and expected frequencies for each patient. Using the formula of expected frequency as follows:

*(E)= (row total * column total) / grand total*

In *Table 3.6*, data the expected frequency of the patient where smoking is yes and lung cancer is yes, is given as follows:

*E = (100 * 100) / 200 = 50*

Refer to the following *Table 3.7*, 50 is expected frequency:

Smoking	Lung Cancer	No Lung Cancer	Total
Yes	80 (50)	20 (50)	100
No	20 (50)	80 (50)	100
Total	100	100	200

Table 3.7: *Expected frequency of patients*

Further to calculate the test statistic, which is the chi-square value. The formula for the chi-square value, is as follows:

$X2 = \Sigma (O-E)2 / E$

Here, O is the observed frequency and E is the expected frequency. The sum is taken over all patient in the *Table 3.6*. For example, the contribution of the *Table 3.6* patient where smoking is yes and lung cancer is yes to the chi-square value is as follows:

(80-50)2 / 50 = 18

The following table shows the contribution of each patient to the chi-square value:

Smoking	Lung Cancer	No Lung Cancer	Total
Yes	18	18	36
No	18	18	36
Total	36	36	72

Table 3.8: *Chi-square value of patients*

Using an alpha value of 0.05 and a degree of freedom of 1, because we have two categories of smoking (yes or no) so *2-1=1*. the critical value from the chi-square distribution table is 3.841. In this case, the test statistic is 72, which is greater than the critical value of 3.841. Therefore, we reject the null hypothesis and conclude that there is a significant association between smoking and lung cancer. This indicates that smoking is a risk factor for lung cancer, making individuals who smoke more susceptible to developing lung cancer compared to non-smokers. This is a straightforward example of how a measure of association can aid in comprehending the relationship between two variables and drawing conclusions about their causal effects.

Covariance and correlation

Covariance is a method for assessing the link between two things. It displays if those two things change in the same or opposite direction. For example, we can use covariance to explore if taller people weigh more or less than shorter people when we investigate whether height and weight are correlated. Let us look at a simple demonstration of covariance. Consider a group of students who take math and English exams. Calculating the relationship between math scores and English scores can tell us if there is a connection between the two subjects. If the covariance is positive, it means that students who excel in math generally perform well in English, and vice versa. If the covariance is

negative, it suggests that students who excel in math usually struggle in English, and vice versa. If the correlation is zero, there is no direct link between math and English scores.

Let us have a look at the following table:

Student	Math score	English score
A	80	90
B	70	80
C	60	70
D	50	60
E	40	50

Table 3.9: *Group of students and their respective grades in Math and English*

Use the formula to compute covariance,

$$Covariance = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{n}$$

Where , x_i, and y_i are the individual scores for math and English, $\bar{x}$ and $\bar{y}$ are the mean scores for math and English, and n is the number of students.

Using the data from Table 3.9, the mean ($\bar{x}$) is 60 and the mean ($\bar{y}$) is 70. The sum of the products of paired deviations $\sum(xi-\bar{x})(yi-\bar{y})$ is 1000. Finally, the covariance between column maths and English score is calculated to be 250. Which means, there is positive linear relation between a student's math and English scores. Their meaning that as one variable increases, the other variable also tends to increase.

Tutorial 3.8: An example to compute the covariance in data, is as follows:

```
import pandas as pd
# Define the dataframe as a dictionary
df = {"Student": ["A", "B", "C", "D", "E"], "Math Score": [
    80, 70, 60, 50, 40], "English Score": [90, 80, 70, 60, 5
0]}
# Convert the dictionary to a pandas dataframe
df = pd.DataFrame(df)
# Calculate the covariance between math and english sc
ores using the cov method
covariance = df["Math Score"].cov(df["English Score"])
# Print the result
print(f"The covariance between math and english score i
s {covariance}")
```

Output:

```
The covariance between math and english score is 250.0
```

Covariance and correlation are similar, but not the same. They both measure the relationship between two variables, but they differ in how they scale and interpret the results.

Following are some key differences between covariance and correlation:

- Covariance can take any value from negative infinity to positive infinity, while correlation ranges from -1 to 1. This means that correlation is a normalized and standardized measure of covariance, which makes it easier to compare and interpret the strength of the relationship.
- Covariance has units, which depend on the units of the two variables. Correlation is dimensionless, which means it has no units. This makes correlation independent of the scale and units of the variables, while covariance is sensitive to them.
- Covariance only indicates the direction of the linear relationship between two variables, such as positive,

negative, or zero. Correlation also indicates the direction, but also the degree of how closely the two variables are related. A correlation of -1 or 1 means a perfect linear relationship, while a correlation of 0 means no linear relationship.

Tutorial 3.9: An example to compute the correlation in the Math and English score data, is as follows:

```
import pandas as pd
# Create a dictionary with the data
data = {"Student": ["A", "B", "C", "D", "E"],
        "Math Score": [80, 70, 60, 50, 40],
        "English Score": [90, 80, 70, 60, 50]}
df = pd.DataFrame(data)
# Compute the correlation between the two columns
correlation = df["Math Score"].corr(df["English Score"])
print("Correlation between math and english score:", correlation)
```

Output:

```
Correlation between math and english score: 1.0
```

Chi-square

Chi-square tests if there is a significant connection between two categories. For example, to determine if there is a connection between the music individuals listen to and their emotional state, chi-squared association tests can be used to compare observed frequencies of different moods with different types of music to expected frequencies if there is no relationship between music and mood. The test finds the chi-squared value by adding the squared differences between the observed and expected frequencies and then dividing that sum by the expected frequencies. If the chi-squared value is higher, it suggests a stronger likelihood of a significant connection between the variables.

The next step confirms the significance of the chi-squared value by comparing it to a critical value from a table that considers the degree of freedom and level of significance. If the chi-squared value is higher than the critical value, we will discard the assumption of no relationship.

Tutorial 3.10: An example to show the use of chi-square test to find association between different types of music and mood of a person, is as follows:

```
import pandas as pd
# Import chi-
squared test function from scipy.stats module
from scipy.stats import chi2_contingency
# Create a sample data frame with music and mood cate
gories
data = pd.DataFrame({"Music": ["Rock", "Pop", "Jazz", "
Classical", "Rap"],
                     "Happy": [25, 30, 15, 10, 20],
                     "Sad": [15, 10, 20, 25, 30],
                     "Angry": [10, 15, 25, 30, 15],
                     "Calm": [20, 15, 10, 5, 10]})
# Print the original data frame
print(data)
# Perform chi-square test of association
chi2, p, dof, expected = chi2_contingency(data.iloc[:, 1:]
)
# Print the chi-square test statistic, p-
value, and degrees of freedom
print("Chi-square test statistic:", chi2)
print("P-value:", p)
print("Degrees of freedom:", dof)
# Print the expected frequencies
print("Expected frequencies:")
```

```
print(expected)
```

Output:

```
     Music  Happy  Sad  Angry  Calm
0      Rock     25   15     10    20
1       Pop     30   10     15    15
2      Jazz     15   20     25    10
3  Classical     10   25     30     5
4       Rap     20   30     15    10
Chi-square test statistic: 50.070718462823734
P-value: 1.3577089704505725e-06
Degrees of freedom: 12
Expected frequencies:
[[19.71830986 19.71830986 18.73239437 11.83098592]
 [19.71830986 19.71830986 18.73239437 11.83098592]
 [19.71830986 19.71830986 18.73239437 11.83098592]
 [19.71830986 19.71830986 18.73239437 11.83098592]
 [21.12676056 21.12676056 20.07042254 12.67605634]
]
```

The chi-square test results indicate a significant connection between the type of music and the mood of listeners. This suggests that the observed frequencies of different music-mood combinations are not random occurrences but rather signify an underlying relationship between the two variables. A higher chi-square value signifies a greater disparity between observed and expected frequencies. In this instance, the chi-square value is 50.07, a notably large figure. Given that the p-value is less than 0.05, we can reject the null hypothesis and conclude that there is indeed a significant association between music and mood. The degrees of freedom, indicating the number of independent categories in the data, is calculated as *(number of rows - 1) x (number of columns - 1)*, resulting in 12 degrees of freedom in this case. Expected frequencies represent what

would be anticipated under the null hypothesis of no association, calculated by multiplying row and column totals and dividing by the grand total. Comparing observed and expected frequencies reveals the expected distribution if music and mood were independent. Notably, rap and sadness are more frequent than expected (30 vs 21.13), suggesting that rap music is more likely to induce sadness. Conversely, classical and calm are less frequent than expected (5 vs 11.83), indicating that classical music is less likely to induce calmness.

Cramer’s V

Cramer's V is a measure of the strength of the association between two categorical variables. It ranges from 0 to 1, where 0 indicates no association and 1 indicates perfect association. Cramer's V and chi-square is related but are different concepts. Cramer's V is an effect size that describes how strongly two variables are related, while chi-square is a test statistic that evaluates whether the observed frequencies are different from the expected frequencies. Cramer's V is based on chi-square, but also takes into account the sample size and the number of categories. Cramer's V is useful for comparing the strength of association between different tables with different numbers of categories. Chi-square can be used to test whether there is a significant association between two nominal variables, but it does not tell us how strong or weak that association is. Cramer's V can be calculated from the chi-squared value and the degrees of freedom of the contingency table.

$$Cramer's\ V = \sqrt{(X2/n)} / min(c-1, r-1)$$

Where:

- **X2**: The Chi-square statistic
- **n**: Total sample size
- **r:** Number of rows

- **c:** Number of columns

For example, Cramer's V is to compare the association between gender and eye color in two different populations. Suppose we have the following data:

Population	Gender	Eye color	Frequency
A	Male	Blue	10
A	Male	Brown	20
A	Female	Blue	15
A	Female	Brown	25
B	Male	Blue	5
B	Male	Brown	25
B	Female	Blue	25
B	Female	Brown	5

Table 3.10: *Gender and eye color in two different populations*

Tutorial 3.11: An example to illustrate the use of Cramer's V to measure the strength of the association between gender and eye color in each population, is as follows:

```
1. import pandas as pd
2. # Importing necessary functions from the scipy.stats mo
   dule
3. from scipy.stats import chi2_contingency, chi2
4. # Create a dataframe from the given data
5. df = pd.DataFrame({"Population": ["A", "A", "A", "A", "B",
    "B", "B", "B"],
6.                    "Gender": ["Male", "Male", "Female", "Femal
   e", "Male", "Male", "Female", "Female"],
7.                    "Eye Color": ["Blue", "Brown", "Blue", "Brow
   n", "Blue", "Brown",
   "Blue", "Brown"],
```

```
                "Frequency": [10, 20, 15, 25, 5, 25, 25, 5]})
# Pivot the dataframe to get a contingency table
table = pd.pivot_table(
    df, index=["Population", "Gender"], columns="Eye Color", values="Frequency")
# Print the table
print(table)
# Perform chi-square test for each population
for pop in ["A", "B"]:
    # Subset the table by population
    subtable = table.loc[pop]
    # Calculate the chi-square statistic, p-value, degrees of freedom, and expected frequencies
    chi2_stat, p_value, dof, expected = chi2_contingency(subtable)
    # Print the results
    print(f"\nChi-square test for population {pop}:")
    print(f"Chi-square statistic = {chi2_stat:.2f}")
    print(f"P-value = {p_value:.4f}")
    print(f"Degrees of freedom = {dof}")
    print(f"Expected frequencies:")
    print(expected)
# Calculate Cramer's V for population B and population A
# Cramer's V is the square root of the chi-square statistic divided by the sample size and the minimum of the row or column dimensions minus one
n = df["Frequency"].sum()  # Sample size
k = min(table.shape) - 1  # Minimum of row or column dimensions minus one
# Chi-square statistic for population B
```

```
chi2_stat_B = chi2_contingency(table.loc["B"])[0]
# Chi-square statistic for population A
chi2_stat_A = chi2_contingency(table.loc["A"])[0]
cramers_V_B = (chi2_stat_B / (n * k)) ** 0.5  # Cramer's V for population B
cramers_V_A = (chi2_stat_A / (n * k)) ** 0.5  # Cramer's V for population A
# Print the results
print(f"\nCramer's V for population B and population A:")
print(f"Cramer's V for population B = {cramers_V_B:.2f}")
print(f"Cramer's V for population A = {cramers_V_A:.2f}")
```

Output:

```
Eye Color          Blue  Brown
Population Gender
A          Female    15     25
           Male      10     20
B          Female    25      5
           Male       5     25

Chi-square test for population A:
Chi-square statistic = 0.01
P-value = 0.9140
Degrees of freedom = 1
Expected frequencies:
[[14.28571429 25.71428571]
 [10.71428571 19.28571429]]

Chi-square test for population B:
Chi-square statistic = 24.07
```

```
P-value = 0.0000
Degrees of freedom = 1
Expected frequencies:
[[15. 15.]
 [15. 15.]]

Cramer's V for population B and population A:
Cramer's V for population B = 0.43
Cramer's V for population A = 0.01
```

Above data shows the frequencies of eye color by gender and population for two populations, A and B. Here, the chi-square test is used to test whether there is a significant association between gender and eye color in each population. The null hypothesis is that there is no association, and the alternative hypothesis is that there is an association. The p-value is the probability of obtaining the observed or more extreme results under the null hypothesis. A small p-value (usually less than 0.05) indicates strong evidence against the null hypothesis, and a large p-value (usually greater than 0.05) indicates weak evidence against the null hypothesis. The results show that for population A, the p-value is 0.9140, which is very large. This means that we fail to reject the null hypothesis and conclude that there is no significant association between gender and eye color in population A. The chi-square statistic is 0.01, which is very small and indicates that the observed frequencies are very close to the expected frequencies under the null hypothesis. The expected frequencies are 14.29 and 25.71 for blue and brown eyes respectively for females, and 10.71 and 19.29 for blue and brown eyes respectively for males. The results show that for population B, the p-value is 0.0000, which is very small. This means that we reject the null hypothesis and conclude that there is a significant association between gender and eye

color in population B. The chi-square statistic is 24.07, which is very large and indicates that the observed frequencies are very different from the expected frequencies under the null hypothesis. The expected frequencies are 15 and 15 for both blue and brown eyes for both females and males.

Since Cramer's V is a measure of the strength of the association between two categorical variables based on the chi-squared statistic and sample size.

The results show that Cramer’s V for population B is 0.43, which indicates a moderate association between gender and eye color. Cramer’s V for population A is 0.01, which indicates a very weak association between gender and eye color. This confirms the results of the chi-square test.

Contingency coefficient

The contingency coefficient is a measure of association in statistics that indicates whether two variables or data sets are independent or dependent on each other. It is also known as Pearson's coefficient.

The contingency coefficient is based on the chi-square statistic and is defined by the following formula:

$$C=\chi 2+N\chi 2$$

Where:

- $\chi 2$ is the chi-square statistic
- N is the total number of cases or observations in our analysis/study.
- C is the contingency coefficient

The contingency coefficient can range from 0 (no association) to 1 (perfect association). If C is close to zero (or equal to zero), you can conclude that your variables are independent of each other; there is no association between them. If C is away from zero, there is some association. Contingency coefficient is important because it can help us

summarize the relationship between two categorical variables in a single number. It can also help us compare the degree of association between different tables or groups.

Tutorial 3.12: An example to measure the association between two categorical variables gender and product using contingency coefficient, is as follows:

```
import pandas as pd
from scipy.stats import chi2_contingency
# Create a simple dataframe
data = {'Gender': ['Male', 'Female', 'Female', 'Male', 'Male', 'Female'],
        'Product': ['Product A', 'Product B', 'Product A', 'Product A', 'Product B', 'Product B']}
df = pd.DataFrame(data)
# Create a contingency table
contingency_table = pd.crosstab(df['Gender'], df['Product'])
# Perform Chi-Square test
chi2, p, dof, expected = chi2_contingency(contingency_table)
# Calculate the contingency coefficient
contingency_coefficient = (chi2 / (chi2 + df.shape[0])) ** 0.5
print('Contingency Coefficient is:', contingency_coefficient)
```

Output:

```
Contingency Coefficient is: 0.0
```

In this case, the contingency coefficient is 0 which shows there is no association at all between gender and product.

Tutorial 3.13: Similarly, as shown in *Table 3.9*, if we want to know whether gender and eye color are related in two

different populations, we can calculate the contingency coefficient for each population and see which one has a higher value. A higher value indicates a stronger association between the variables.

Code:

```
import pandas as pd
from scipy.stats import chi2_contingency
import numpy as np
df = pd.DataFrame({"Population": ["A", "A", "A", "A", "B", "B", "B", "B"],
                   "Gender": ["Male", "Male", "Female", "Female", "Male", "Male", "Female", "Female"],
                   "Eye Color": ["Blue", "Brown", "Blue", "Brown",
"Blue", "Brown", "Blue", "Brown"],
                   "Frequency": [10, 20, 15, 25, 5, 25, 25, 5]})
# Create a pivot table
pivot_table = pd.pivot_table(df, values='Frequency', index=[
                             'Population', 'Gender'], columns=['Eye Color'], aggfunc=np.sum)
# Calculate chi-square statistic
chi2, _, _, _ = chi2_contingency(pivot_table)
# Calculate the total number of observations
N = df['Frequency'].sum()
# Calculate the Contingency Coefficient
C = np.sqrt(chi2 / (chi2 + N))
print(f"Contingency Coefficient: {C}")
```

Output:

```
Contingency Coefficient: 0.43
```

This gives contingency coefficient 0.4338. Which indicates that there is a moderate association between the variables

in the above data (population, gender, and eye color). This means that knowing the category of one variable gives some information about the category of the other variables. However, the association is not very strong because the coefficient is closer to 0 than to 1. Furthermore, the contingency coefficient has some limitations, such as being affected by the size of the table and not reaching 1 for perfect association. Therefore, some alternative measures of association, such as Cramer's V or the phi coefficient, may be preferred in some situations.

Measures of shape

Measures of shape are used to describe the general shape of a distribution, including its symmetry, skewness, and kurtosis. These measures help to give a sense of how the data is spread out, and can be useful for identifying potentially outlier observations or data points. For example, imagine you are a teacher, and you want to evaluate your students' performance on a recent math test. Here the skewness tells you distribution of the scores. If the scores are more spread out on one side of the mean than the other, and kurtosis tells you how peaked or flattened the distribution of scores is.

Skewness

Skewness measures the degree of asymmetry in a distribution. A distribution is symmetrical if the two halves on either side of the mean are mirror images of each other. Positive skewness indicates that the right tail of the distribution is longer or thicker than the left tail, while negative skewness indicates the opposite.

Tutorial 3.14: Let us consider a class of 10 students who recently took a math test. Their scores (out of 100) are as follows, and based on these scores we can see the skewness of the students' scores, whether they are positively skewed

(toward high scores) or negatively skewed (toward low scores).

Refer to the following table:

Student ID	1	2	3	4	5	6	7	8	9	10
Score	85	90	92	95	96	96	97	98	99	100

Table 3.11: *Students and their respective scores*

Code:

```
1. import matplotlib.pyplot as plt
2. import seaborn as sns
3. from scipy.stats import skew
4. data = [85, 90, 92, 95, 96, 96, 97, 98, 99, 100]
5. # Calculate skewness
6. data_skewness = skew(data)
7. # Create a combined histogram and kernel density plot
8. plt.figure(figsize=(8, 6))
9. sns.histplot(data, bins=10, kde=True, color='skyblue', e
   dgecolor='black')
10. # Add skewness information
11. plt.xlabel('Score')
12. plt.ylabel('Count')
13. plt.title(f'Skewness: {data_skewness:.2f}')
14. # Show the figure
15. plt.savefig('skew_negative.jpg', dpi=600, bbox_inches='t
    ight')
16. plt.show()
```

Output: *Figure 3.3* shows negative skew:

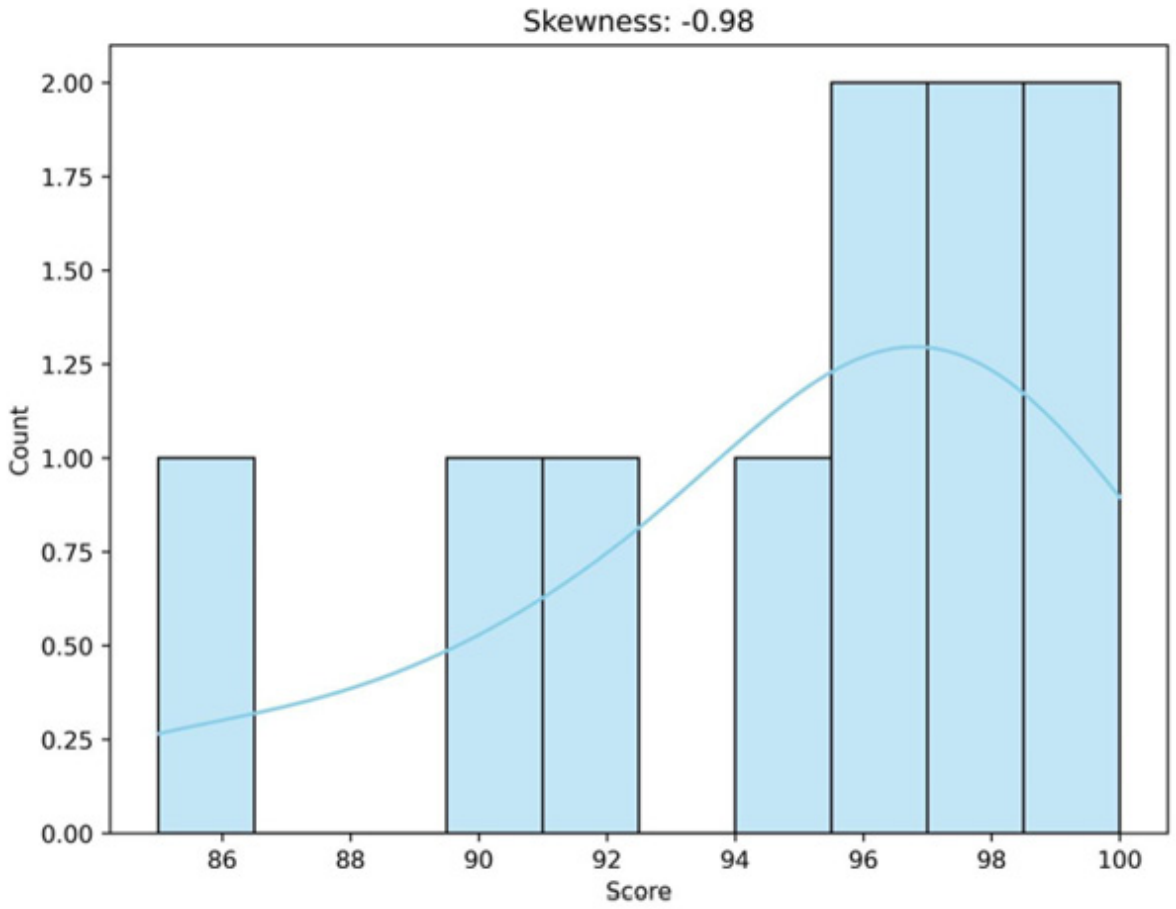

Figure 3.3: *Negative skewness*

The given data, exhibiting a skewness of -0.98, is negatively skewed. The graphical representation indicates that the distribution of students; scores is not symmetrical. The majority of scores are concentrated to the left, while fewer scores are concentrated to the right. This is an example of **negative skewness**, also known as left skew. In a negatively skewed distribution, the mean is smaller than the median, and the left tail (smaller numbers) is longer or thicker than the right tail. In this scenario, the teacher can deduce that most students scored below the average on the test, with very few scorings above the average. This could suggest that the test was challenging or that the class faces difficulties in the subject matter. Remember that skewness is only one aspect of understanding the distribution of data.

It is also important to consider other factors, such as kurtosis, standard deviation, etc., for a more complete understanding.

Tutorial 3.15: An example to view the positive skewness of data, is as follows:

```
import matplotlib.pyplot as plt
import seaborn as sns
from scipy.stats import skew
data = [115, 120, 85, 90, 92, 95, 96, 96, 97, 98]
# Calculate skewness
data_skewness = skew(data)
# Create a combined histogram and kernel density plot
plt.figure(figsize=(8, 6))
sns.histplot(data, bins=10, kde=True, color='skyblue', edgecolor='black')
# Add skewness information
plt.xlabel('Score')
plt.ylabel('Count')
plt.title(f'Skewness: {data_skewness:.2f}')
# Display the plot
plt.savefig('skew_positive.jpg', dpi=600, bbox_inches='tight')
plt.show()
```

Output: *Figure 3.4* shows positive skew:

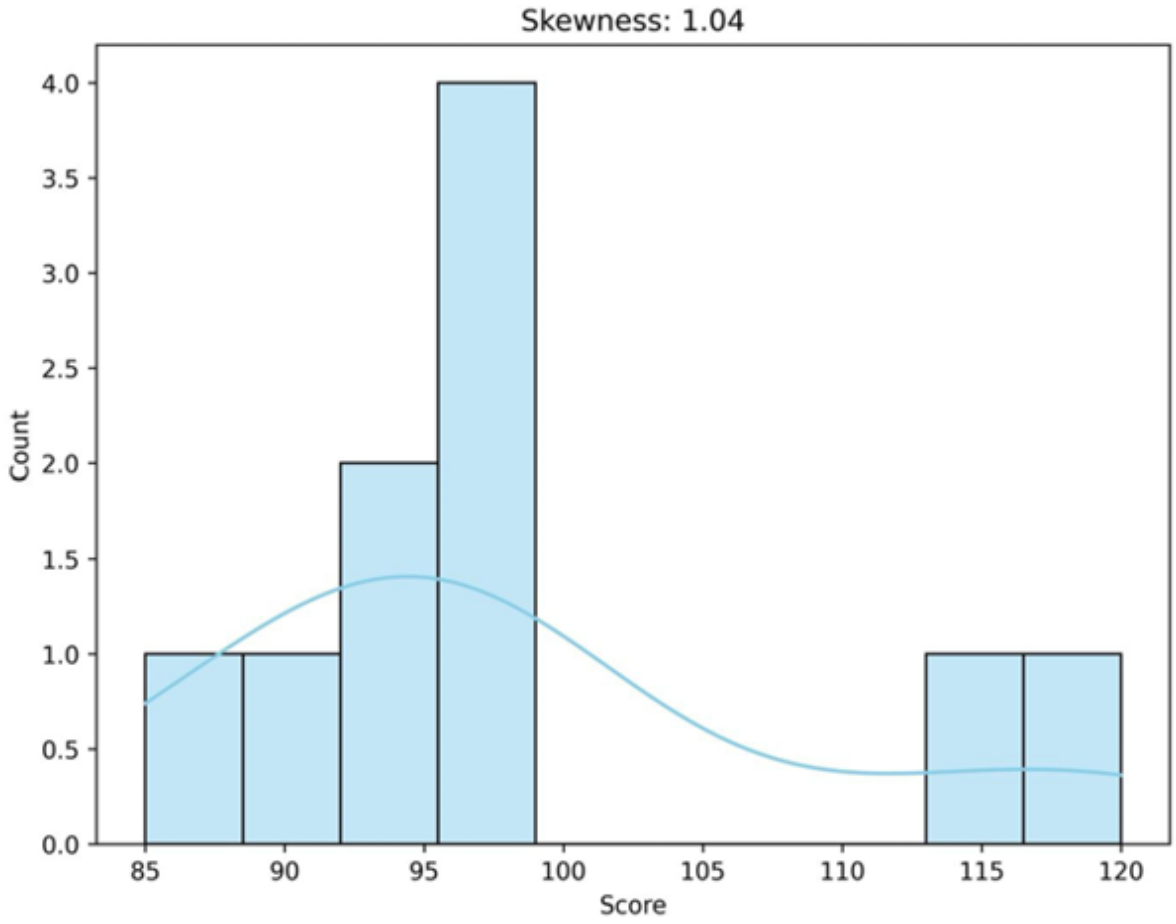

Figure 3.4: *Positive skewness*

Tutorial 3.16: An example to show the symmetrical distribution, positive and negative skewness of data respectively in a subplot, is as follows:

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import skew
# Define the three datasets
data1 = np.array([1, 2, 3, 4, 5, 5, 4, 3, 2, 1])
data2 = np.array([2, 3, 4, 5, 6, 7, 8, 9, 10, 20])
data3 = np.array([20, 15, 10, 9, 8, 7, 6, 5, 4, 2])
# Calculate skewness for each dataset
skewness1 = skew(data1)
```

```
skewness2 = skew(data2)
skewness3 = skew(data3)
# Plot the data and skewness in subplots
fig, axes = plt.subplots(1, 3, figsize=(12, 8))
# Subplot 1
axes[0].plot(data1, marker='o', linestyle='-')
axes[0].set_title(f'Data 1\nSkewness: {skewness1:.2f}')
# Subplot 2
axes[1].plot(data2, marker='o', linestyle='-')
axes[1].set_title(f'Data 2\nSkewness: {skewness2:.2f}')
# Subplot 3
axes[2].plot(data3, marker='o', linestyle='-')
axes[2].set_title(f'Data 3\nSkewness: {skewness3:.2f}')
# Adjust layout
plt.tight_layout()
# Display the plot
plt.savefig('skew_all.jpg', dpi=600, bbox_inches='tight')
plt.show()
```

Output:

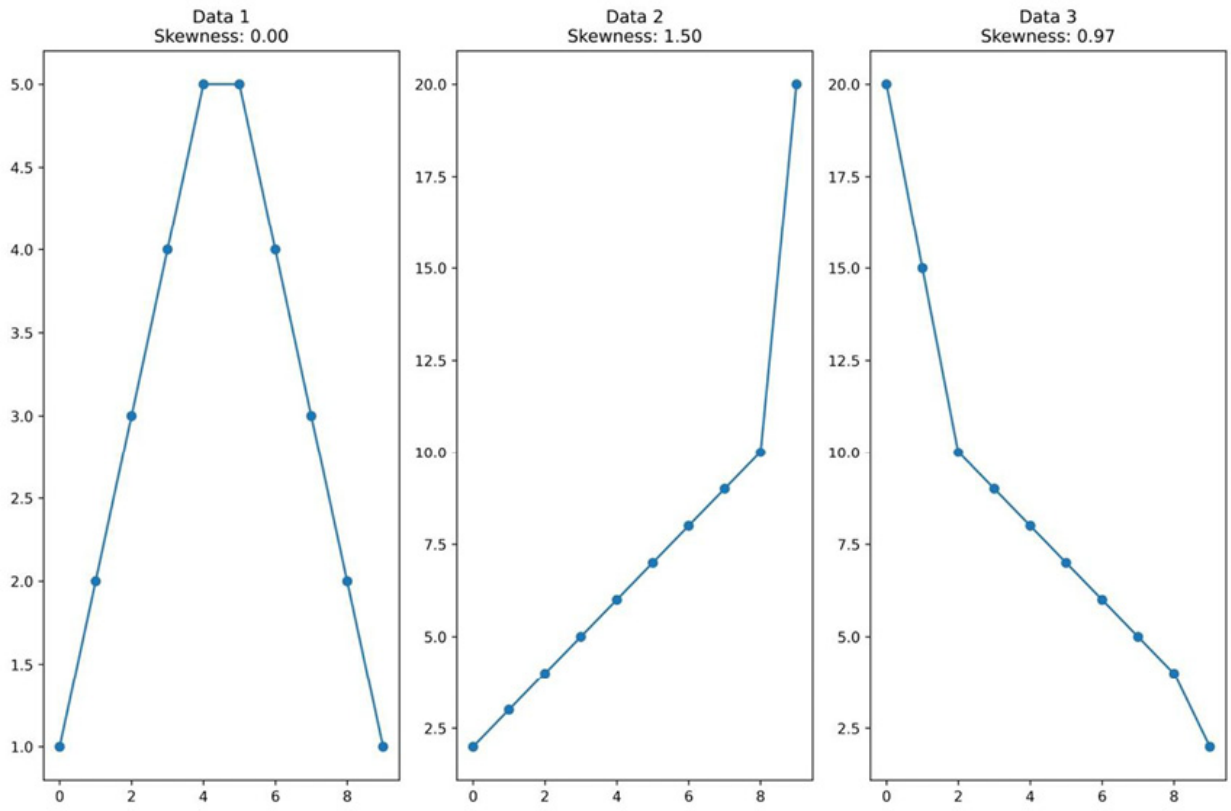

Figure 3.5: *Symmetrical distribution, positive and negative skewness of data*

Tutorial 3.17: An example to measure skewness in diabetes dataset data frame **Age** column using plot, is as follows:

```
1. import pandas as pd
2. import matplotlib.pyplot as plt
3. import seaborn as sns
4. from scipy.stats import skew
5. diabities_df = pd.read_csv(
6.     '/workspaces/ImplementingStatisticsWithPython/data
   /chapter1/diabetes.csv')
7. data = diabities_df['Age']
8. # Calculate skewness
9. data_skewness = skew(data)
10. # Create a combined histogram and kernel density plot
11. plt.figure(figsize=(8, 6))
12. sns.histplot(data, bins=10, kde=True, color='skyblue', e
```

```
    dgecolor='black')
# Add skewness information
plt.title(f'Skewness: {data_skewness:.2f}')
# Display the plot
plt.savefig('skew_age.jpg', dpi=600, bbox_inches='tight')
plt.show()
```

Output:

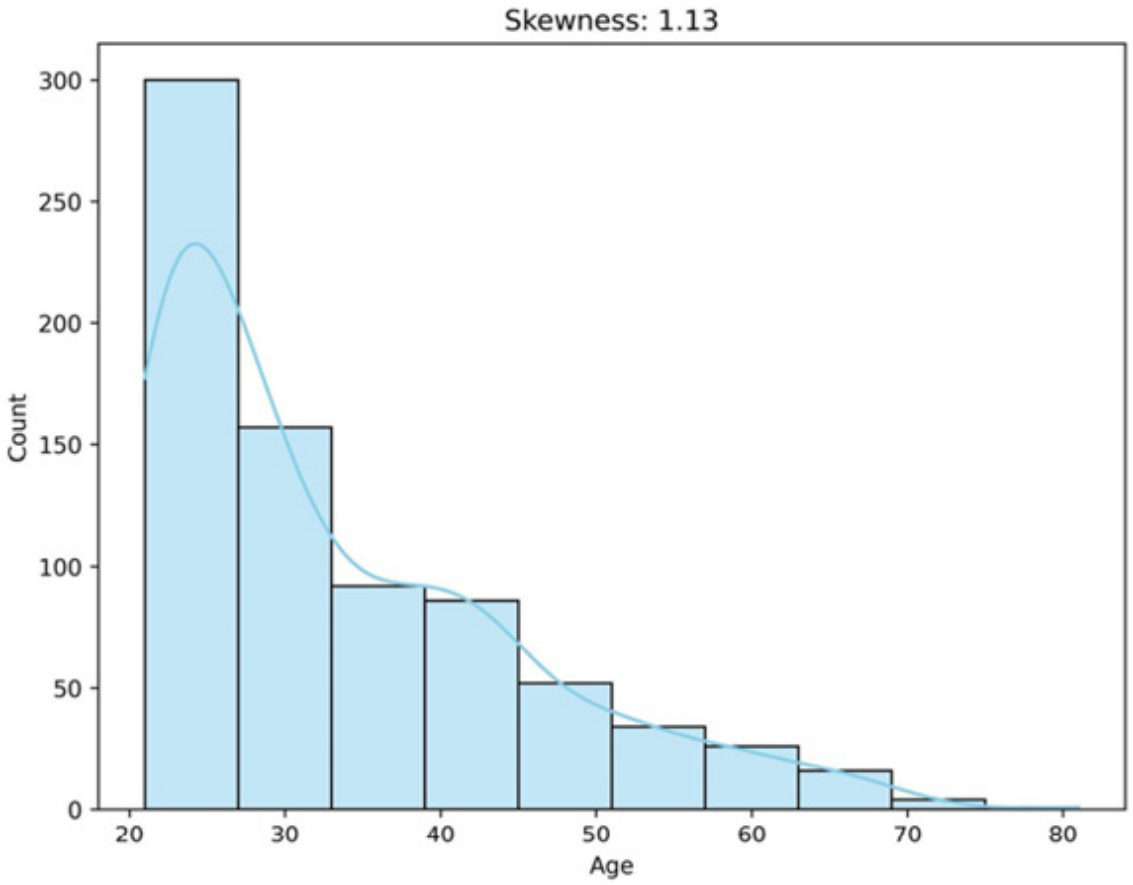

Figure 3.6: *Positive skewness in diabetes dataset Age column*

Kurtosis

Kurtosis measures the tilt of a distribution (that is, the concentration of values at the tails). It indicates whether the tails of a given distribution contain extreme values. If you think of a data distribution as a mountain, the kurtosis

would tell you about the shape of the peak and the tails. A high kurtosis means that the data has heavy tails or outliers. In other words, the data has a high peak (more data in the middle) and fat tails (more extreme values). This is called a **leptokurtic distribution**. Low kurtosis in a data set is an indicator that the data has light tails or lacks outliers. The data points are moderately spread out (less in the middle and less extreme values), which means it has a flat peak. This is called a **platykurtic distribution**. A normal distribution has zero kurtosis. Understanding the kurtosis of a data set helps to identify volatility, risk, or outlier detection in various fields such as finance, quality control, and other statistical modeling where data distribution plays a key role.

Tutorial 3.15: An example to understand how viewing the Kurtosis of a dataset helps in identifying the presence of outliers.

Let us look at three different data sets, as follows:

- **Dataset A:** [1, 1, 2, 2, 3, 3, 4, 4, 9, 9] - This dataset has a few extreme values (9).
- **Dataset B:** [1, 2, 3, 4, 5, 6, 7, 8, 9, 10] - This dataset has no extreme values and is evenly distributed.
- **Dataset C:** [1, 2, 2, 3, 3, 3, 4, 4, 4, 4] - This data set has more values around the mean (3 and 4).

Let us calculate the kurtosis for these data sets.

Code:

```
import scipy.stats as stats
# Datasets
dataset_A = [1, 1, 2, 2, 3, 3, 4, 4, 4, 30]
dataset_B = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
dataset_C = [1, 2, 3, 3, 3, 3, 3, 3, 4, 5]
# Calculate kurtosis
kurtosis_A = stats.kurtosis(dataset_A)
```

```
kurtosis_B = stats.kurtosis(dataset_B)
kurtosis_C = stats.kurtosis(dataset_C)
print(f"Kurtosis of Dataset A: {kurtosis_A}")
print(f"Kurtosis of Dataset B: {kurtosis_B}")
print(f"Kurtosis of Dataset C: {kurtosis_C}")
```

Output:

```
Kurtosis of Dataset A: 4.841818043320611
Kurtosis of Dataset B: -1.2242424242424244
Kurtosis of Dataset C: 0.3999999999999999
```

Here we see, in data set *A: [1, 1, 2, 2, 3, 3, 4, 4, 4, 30]* has a kurtosis of 4.84. This is a high positive value, indicating that the data set has heavy tails and a sharp peak. This means that there are more extreme values in the data set, as indicated by the value 30. This is an example of a leptokurtic distribution. In the data set *B: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]* has a kurtosis of -1.22. This is a negative value, indicating that the data set has light tails and a flat peak. This means that there are fewer extreme values in the data set and the values are evenly distributed. This is an example of a platykurtic distribution. The data set *C: [1, 2, 3, 3, 3, 3, 3, 3, 3, 4, 5]* has a kurtosis of 0.4, which is close to zero. This indicates that the data set has a distribution shape similar to a normal distribution (mesokurtic). The values are somewhat evenly distributed around the mean, with a balance between extreme values and values close to the mean.

Conclusion

Descriptive statistics is a branch of statistics that organizes, summarizes, and presents data in a meaningful way. It uses different types of measures to describe various aspects of the data. For example, measures of frequency, such as relative and cumulative frequency, frequency tables and

distribution, help to understand how many times each value of a variable occurs and what proportion it represents in the data. Measures of central tendency, such as mean, median, and mode, help to find the average or typical value of the data. Measures of variability or dispersion, such as range, variance, standard deviation, and interquartile range, help to measure how much the data varies or deviates from the center. Measures of association, such as correlation and covariance, help to examine how two or more variables are related to each other. Finally, measures of shape, such as skewness and kurtosis, help to describe the symmetry and the heaviness of the tails of a probability distribution. These methods are vital in descriptive statistics because they give a detailed summary of the data. This helps us understand how the data behaves, find patterns, and make knowledgeable choices. They are fundamental for additional statistical analysis and hypothesis testing.

In *Chapter 4: Unravelling Statistical Relationships* we will see more about the statistical relationship and understand the meaning and implementation of covariance, correlation and probability distribution.

Join our book’s Discord space

Join the book's Discord Workspace for Latest updates, Offers, Tech happenings around the world, New Release and Sessions with the Authors:

https://discord.bpbonline.com